

# The Story Behind the High Failure Rates in the IT Sector

*By R. Goatham – Calleam Consulting Ltd.*

The fact that project failures rates in the IT industry are considerably higher than for other types of engineering projects has been well documented. Over the years many studies have shown high failure rates in the IT sector [1, 2] and among the latest reports is a Jul 2008 study by US Government Accounting Office (GAO) that found that of 840 federally funded projects 49% of were poorly planned, poorly performing or both [3]. While some would like to believe that the situation is better in private organizations a 2008 study by the Information Systems Audit and Control Association that found that 43% of 400 respondents admitted that their organization had had a recent project failure [4].

Clearly failure rates in fields such as civil engineering are much lower. While construction projects do at times go over budget and encounter schedule delays, the majority of projects do get completed and the end product usually fulfils its intended purpose. If the failure rates experienced in the IT sector were replicated in civil engineering projects our cities would be littered with abandoned construction projects, the electrical supply to our homes would work intermittently and many of our bridges would have gaping holes that would routinely swallow vehicles brave enough to attempt a crossing.

Given that IT Project Managers use the same basic tools, techniques and principles as their counterparts in civil engineering, the difference in success rates raises some interesting questions.

- 1) Are IT projects inherently different from other types of projects such as those found in the civil engineering sector?
- 2) Is the difference in success rates just a reflection of the fact that the application of Project Management principles in the IT sector has not attained the same level of maturity as it has in the civil engineering?
- 3) Are there environmental factors in the IT sector that are poorly recognized and hence poorly managed?

Superficially, all three are probably true, however, if we want to improve the success rates in the IT sector, we need to go beyond the superficial answers and understand the problems in a much more fundamental way. There is an old proverb that says that “you can’t *truly* solve a problem until you *truly* understand the nature of the problem you’re solving”. In short, when we fail to understand the essence of a problem, we run the risk of adopting solutions that simply don’t work.

Many IT organizations have been down that road. In the aftermath of a failed project, one of the most common reactions is for the organization to adopt a program of process improvement. In theory by defining more robust processes for the planning, execution and control of projects, success rates can be improved. Having witnessed many organizations attempt process improvement I can report that it’s an idea that is far harder to implement than it might first appear to be and a good number of the organizations that attempt process improvement end up no better off than they had before the

program began. The reasons behind the failure of process improvement initiatives are often complex, but in large part it often comes down to the fact that the underlying problems the organization was facing were poorly understood and hence the process improvement initiatives was poorly attuned to solving the organizations real needs.

## *Projects from first principles*

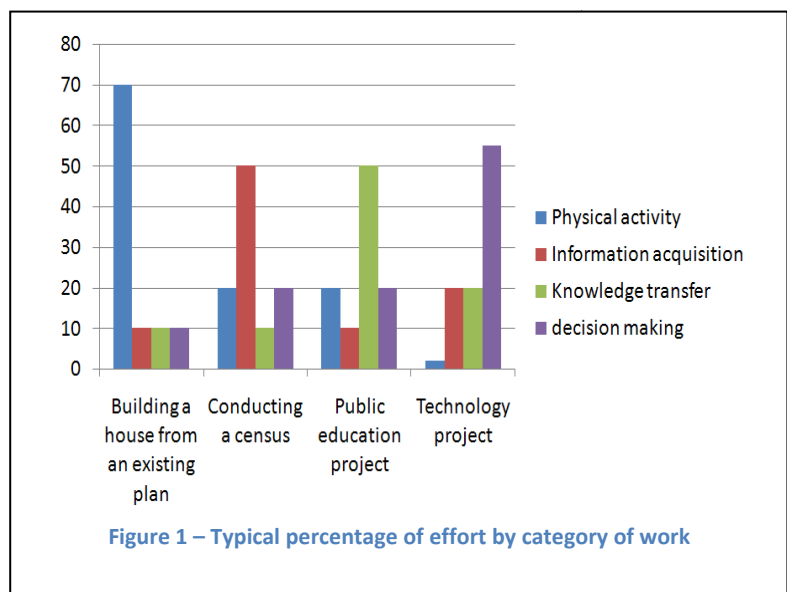
Rather than attempting another round of quality improvement initiatives, perhaps the greatest hope for improving success rates lies in going back to first principles and understanding the very essence of what it is that makes managing IT projects so hard to do. Only then will we be able to develop practices and management approaches that truly overcome the problems.

The starting point for doing that lies in being able to step back from our traditional view of a project so that we can see the work we do at its most fundamental level. Traditionally projects have been thought of as sets of discrete tasks that are linked to each other through dependencies. Most commonly, the tasks and their dependencies are represented using a Gantt chart. Each task is visibly delineated from the next and the relationships between tasks are clearly defined. While a Gantt chart provides a useful visualization of a project, it represents a simplification of reality. Although simplifications are necessary the danger in any simplification is that it obscures the more complex reality. In the case of the IT industry, the gap between the simplified view and reality is significant and the space between the two is the breeding ground for the problems that lead to project failure.

Understanding the complex reality that technology team's face requires us to go beyond the relatively simple "task centric" view and look at projects in an entirely different way. At their most elementary level, all work carried out in a project can be thought of using four general categories;

- 1) Physical tasks (such as laying foundations, building walls, etc
- 2) Information acquisition and analysis
- 3) Information and knowledge transfer
- 4) Decision making

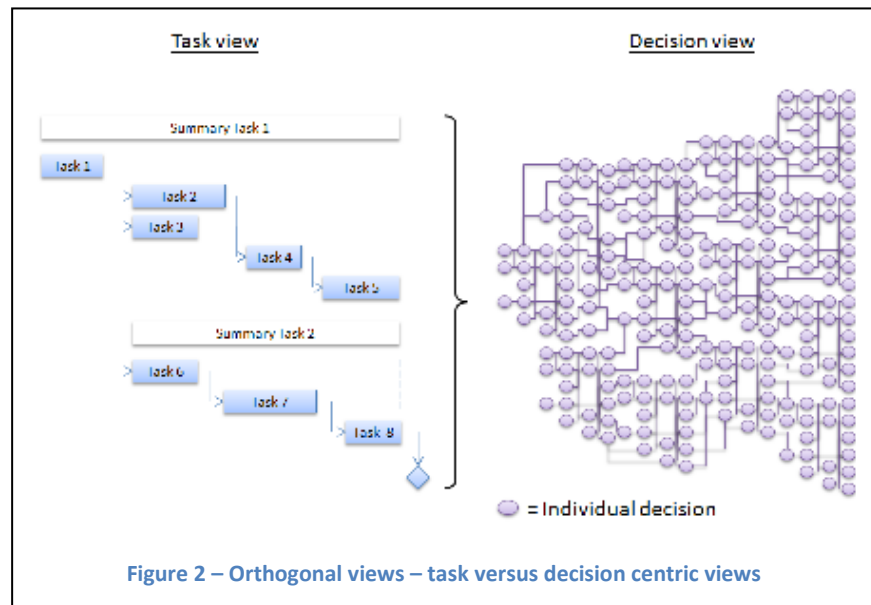
The percentage of effort spent on each activity varies according to the type of project [Figure 1]. While construction projects have a heavy bias towards physical activities,



design type projects are dominated by the making of decisions. Figure 1 provides some representative examples of the different profiles of work that might be seen in different types of projects.

As shown in Figure 1 the work in an IT project is dominated by decision making. Rather than being “physical tasks”, the majority of the work we refer to as “tasks” in a technology project, are in fact “decision making” activities. From strategy development all the way through to developing code decision making is the primary activity that absorbs the majority of effort that goes into an IT project.

Although the “task centric” view and the Gantt chart are firmly entrenched in our thinking about projects, rather than being a set of interdependent tasks, the reality is that IT projects are complex webs of interrelated decisions (Figure 2).



So ubiquitous is decision making that we often don’t think about our work in that way. While teams are generally aware of the key decisions made in a project, few will have considered that decision making is the pervasive act that determines even the smallest detail of a project’s outcome. The central role that decision making plays means that the level of success a project team achieves is directly correlated to the effectiveness of the team’s collective decision-making capabilities. If the team is able to consistently make good decisions the chances are they will succeed. If the team makes too many bad decisions the chances of success are much reduced.

### *Compounding the Challenge*

IT projects cannot claim to be the only decision centric projects and many other types of project fall into the same class. Designing a new car or designing a building represent other examples. There are however a number of factors that makes decision making in the IT sector particularly tricky.

- 1) IT projects usually involve many individuals and as a result the decision making is often heavily decentralized across the team
- 2) Many decisions are subjective and there are no definitively right or wrong answers

- 3) Most decisions lack the physical attributes that more readily lend themselves to visualization, verification and communication (such as height, depth, breadth and mass)
- 4) People with diverse paradigms and perspectives need to work together. Those differing backgrounds compound the difficulty of communicating effectively
- 5) Many decisions in an IT project involve significant uncertainty
- 6) The decisions are often mutually dependent. A decision in one part of the project can have implications across many other parts of the project

The high number of the decisions made in a typical IT project and the complex interactions and dependencies that exist between those decisions, represents many magnitudes of complexity beyond the simple representation of a project captured in a Gantt chart. By putting aside the task centric simplification that has become the standard view of a project, we can say that the fundamental problem that makes managing an IT project so hard to do can be encapsulated by the question; “How do we manage a large scale, complex, decentralized, decision making activity?”

### *Onion Rings*

That central idea provides an anchor point for debate, however much like the rings in an onion there are further layers to the problem. I refer to the next layer as the six great challenges and although the challenges can be expressed in very simple terms, in practice they are a considerable challenge. The six great challenges are;

- 1) Knowing what decisions need to be made and when
- 2) Identifying optimal choices for each decision
- 3) Managing the complexity inherent in the sheer number of decisions needing to be made
- 4) Recognizing and managing the uncertainties inherent in the decisions
- 5) Maintaining the integrity of the whole (i.e. ensuring the compatibility and alignment of all of the decisions made)
- 6) Detecting and eliminating errors in the decisions made

Those six elements alone represent a considerable challenge, but beyond the six great challenges is an even broader context within which project related decisions are made. While it would be comforting to think that all decisions are made in a fully rational and informed way, in practice many other dynamics influence the way decisions are made. Diverse factors such as cognitive biases, training, prior experiences, interpersonal relationships and personality type can shape the decisions made on an individual basis and in the broader context factors such as politics, organizational goals (both spoken and unspoken) and the structure of incentives within the organization can also influence the choices that get made. When other environmental factors that affect decision making (such as the amount of schedule or budget pressure the team is working under) are considered, it becomes clear that the

domain within which decisions are made is an extremely complex one and that complexity brings with it the ever present danger of project failure.

### *Experts and Expertise*

In large part the antidote to mastering the complex domain in which decisions are made is “expertise”. Expertise is of course a critical ingredient in any decision centric activity. Experts have the insight needed to be able to address the six great challenges and because of their experience they are better equipped to navigate the complex domain within which project decisions are made. Their prior experiences give them the situational awareness needed to be able to ask the right questions at the right time and the ability to identify optimal answers. Their depth of understanding reduces the level of uncertainty associated with the decisions they make and also helps them avoid making too many mistakes. Those advantages make experts more productivity than others and that in turn can improve the overall project environment by helping to reduce the stress level the team is exposed to.

Of course everyone likes to think of themselves as an expert. Pick up a stack of resumes and you’ll find the word used liberally. The problem is that as industry the IT sector has generally poorly understood the nature of expertise, the processes by which it develops and how to recognize it when building a team. In most practical situations organizations simply equate years of experience with expertise. However as those in the trenches are fully aware, the difference in capabilities between individuals can be vast. Studies on the subject often show a 10 to 1 variance between the most and least capable IT workers [5, 6]. The net results is that organizations can at times end up with teams whose capabilities fall short of that required to ensure the success of the project and it is that gap which provides the tinder from which project failures occur.

In part the problem is structural to the industry. Unlike other professions that are “decision centric” and “expert” driven (such as medicine, law and engineering) the IT sector lacks a professional infrastructure that establishes and maintains levels of professional practice. While in the legal and medical professions the barriers to entry are high and practitioners can be disbarred if their services fail to meet professional standards, in the IT sector the barriers to entry are low and there are no professional bodies with any form of real authority. Although various bodies do offer certification programs for IT professionals, the certifications in the IT sector are generally toothless and often mean little more than a person has memorized some material from a book.

Interestingly other project environments in which the barriers to entry are low and there are no governance bodies suffer from similar problems as the IT industry. One example is the home renovation business. With the boom in the housing market that took place a few years ago there was a corresponding boom in the need for contractors to do renovations. Again there were no barriers to entry into the sector and no governing bodies to oversee professional practice. Although there are good renovation contractors out there, there are also many who lack the expertise to be doing what they are

doing. As a result complaints about failed renovation projects represent one of the most common complaints reported to the Better Business Bureau [7]. So significant is the problem that here in Canada there is a highly successful television program (Holmes on Homes) in which an expert in home renovation visits the homes of people who have fallen victim to shoddy contractors and helps them fix up the problems. Given that the show is currently in its 7<sup>th</sup> season and runs on networks around the world, it seems that it's an issue that resonates with many people.

### *Barriers to Developing Expertise*

To be fair to the industry, developing expertise in a project based environment is a major challenge. In many ways the project environment is the “antithesis” of the environment people need in order to learn on the job. As a leader in understanding the processes by which people develop expertise, Gary Klein of Klein Associates has shown that people develop expertise fastest when they are in the right environment. That environment requires four elements to be present [8];

- 1) Exposure to repeated patterns of events
- 2) Rapid feedback between a decision being made and the decision's outcome becoming manifest
- 3) The ability to isolate out individual decisions and link them to their outcomes
- 4) Time to reflect on those outcomes

In an IT project based environment there are many obstacles that stand in the way of creating those four conditions. Among other things, the characteristics of IT projects are such that;

- 1) Because projects extend over significant periods of time there is little immediate feedback on the decisions we make
- 2) By definition projects are all unique. That uniqueness makes identifying repeated patterns more challenging
- 3) Projects involve complex webs of interrelated decisions which makes isolating out individual cause and effect relationships hard to do
- 4) Where projects extend over significant periods of time, participants may have only experienced a handful of full project cycles. This again reduces the exposure to repeated patterns
- 5) Project teams are often working under constant pressure so they lack the time to look back and reflect on performance

Given the problems listed above and the fact that many IT development centers spend relatively little money on training, many IT workers learn their skills through the school of hard knocks. Those who have been down that road can attest to how rough a journey that can be. The cost to both the organization and the individual can be high and stories of organizations made bankrupt by a failed project, people leaving the industry because of stress and senior executives fired following a project failure are all too easy to find.

## *Conclusion*

The inherent characteristics of IT projects are such that IT projects will always be complex endeavors. Combined with the low barriers to entry into the profession, the lack of a governing body, the obstacles to developing expertise and the often low levels of investment in training the opportunity for IT projects to go awry is ever present. Throw in rapidly changing technology (which means by the time you become an expert the game has already changed) and constant pressure to deliver (which reduces think time thereby leading to further mistakes) and you have the perfect storm for project failure to occur.

Of course many IT projects do succeed and developing expertise in the IT sector is not impossible. Increasing the chances of project success does however require organizations to think very carefully about how they build teams and the processes they use for ensuring that their teams have the necessary capabilities to succeed. Although it's always tempting to blame the Project Manager or the project team when a project fails, most often failure is a reflection of broader problems within the organization as a whole. Rather than blaming the team, organizations would be better served if they took a long hard look at their management, training and hiring practices. Only then will organizations be able to build the necessary infrastructure within which expertise will flourish.

A key part of the changes organizations need to make comes down to recognizing the decision-centric complexity that is the essence of an IT project. While it's easy to get lured into a false sense of security because a Gantt chart can make an IT project look relatively simply, the underlying web of interrelated decisions is the dimension that makes managing IT projects so hard to do. By recognizing that dimension and understanding the many and varied factors that affect the way individuals, teams and organizations make decisions, organizations can start on the long road towards developing the levels of expertise needed to be able to improve their chances of success.

R. Goatham – Principal Calleam Consulting Ltd.

[Robert@calleam.com](mailto:Robert@calleam.com)

[www.calleam.com](http://www.calleam.com)

About Calleam Consulting: Calleam Consulting provides training and consulting services to the technology sector. Specializing in advanced Project Management training Calleam focuses on helping Project Managers develop the high order thinking skills needed to manage today's complex projects. Using simulation, modeling and analysis of both successful and failed projects from the past, Calleam helps organizations turn yesterday's hindsight into the foresight needed for tomorrow. For more information visit [www.calleam.com](http://www.calleam.com)

## *References*

- [1] Chaos Report – Standish Group
- [2] Dr Dobbs – Defining Success – S. Ambler – Oct 2007
- [3] GAO-08-1051T - United States Government Accountability Office testimony – Jul 2008 (OMB and agencies need to improve planning, management and oversight of projects)
- [4] IT Week Magazine – 19 May 2008
- [5] Software Cost Estimation with Cocomo II – B. Boehm, et al – Addison Wesley, 2000
- [6] Peopleware (productive projects and teams) 2nd edition – T. DeMarco, T. Lister – Dorset House Publishing, 1999
- [7] Better Business Bureau 2008 Complaints Statistics
- [8] Sources of Power (How people make decisions) – G. Klein – MIT Press.